

# **Compact GVRP**

**Norman Finn**

# Compact GVRP Goals

- **Reduce the number of packets required to transmit 4k VLANs' states.**
- **Reduce the number of timers required whenever more than one packet is required to transmit the state of a port's GARP application.**
- **Take advantage of the fact that, on a point-to-point link, **you** are the ultimate authority on what **I** need to register, to speed up convergence.**
- **Work with single packets, if not fast convergence, over shared media.**
- **Maintain full compatibility with standard GVRP.**

# 1. One PDU carries complete GVRP state

- Four per-VLAN messages required for standard GVRP: **JoinIn**, **JoinEmpty**, **Empty**, and **Leave**.
- So, 2 bits times 4k VLANs = 1024 bytes. Therefore, GVRP can transmit a vector of all VLANs' states in one PDU. (Piece of cake!)
- However, there is another state, for which no message is sent. This would be the **In** message, mentioned in 802.1D. When transmitting a vector of states, there is no notion of “no message”. Therefore, there are **five** messages, not four.
- 3 bits times 4k VLANs = 1536 bytes, which is too big!

# 1. One PDU carries complete GVRP state

- However,  $5^{**} 4096 < 256^{**} 1189$ , so we can encode the messages one PDU.
- 3 messages =  $5 * 5 * 5 = 125$  states, which fits in one byte. 13 messages can fit into 4 bytes. At 3 messages per byte, **1366** bytes will carry **4096** messages.

## **2. Reduce the number of timers**

- **GARP requires a separate joinTimer for each VLAN if multiple PDUs must be sent in order to transmit all VLANs' states.**
- **Packing all data into one PDU fixes this.**

### **3. Point-to-point links**

- **On a point-to-point link, if you say you don't want a given VLAN, then I can be sure, without using a timer, that I no longer need to supply that VLAN.**
- **Therefore, I can pass that information on immediately, instead of waiting to see whether some other device on the same LAN wants that VLAN; there are no other devices.**
- **This can result in a large number of PDUs being sent, if I receive data on multiple ports, so we still need an anti-chatter timer.**

### 3. Point-to-point links

- But, how do you **know** that this is a point-to-point link?
- Answer:
  - Include a “This is my system identifier” message in a C-GVRP PDU.
  - Track the number of system identifiers received on a port in order to ensure that point-to-point mode is not used when there are more than two players.
- Switch back to “timer driven” mode, the same as standard GVRP, when on a shared medium.

## **4. Maintain shared media compatibility**

- In shared mode, operate exactly as GVRP operates today, except that the new, condensed packet format can be used.**



# Maintain compatibility with GVRP

- **Divide message vectors into chunks to fit in GARP-standard messages. (This also permits shorter PDUs if only some VLANs are used.)**
- **Send “LeaveAll” along with “JustKidding” global messages to tease out old GVRP implementations.**
- **If any old GVRP implementations present, revert to old GVRP.**
- **Not 100% guaranteed: if a device joins no VLANs, but is able to register them, then it will remain silent, and will not register the Compact GVRP Joins.**
- **Is this really important?**